METHOD AND SYSTEM FOR JOINT DECISION FEEDBACK EQUALIZATION AND COMPLEMENTARY CODE KEY DECODING USING A TRELLIS

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is related to applicant's co-pending United States Patent Application Serial No. 09/107,546 filed June 30, 1998 entitled METHOD AND DEVICE FOR IMPROVING DFE PERFORMANCE IN A TRELLIS-CODED SYSTEM, the whole contents and disclosure of which is incorporated by reference as if fully set forth herein.

BACKGROUND OF THE INVENTION

FIELD OF THE INVENTION

This invention generally relates to digital communication systems, and more specifically, to an enhanced system and method for performing feedback equalization and complementary code key (CCK) decoding using a trellis structure.

PRIOR ART

In many digital communication scenarios (e.g. telephone transmission, broadcast TV transmission, cable etc.). The transmitted signal arrives at the receiver through more than one path in addition to the direct path. This condition is called "multipath" and leads to intersymbol interference ("ISI") in the digital symbol stream. This ISI is compensated for in the receiver

US020010 (702518)

through an equalizer which in many cases is a DFE as shown in Figure 1. U.S. Patent No 5,572,262 shows one method of combating these multipaths.

A DFE 10 (Fig. 1) has two filter sections, a forward filter 12 and a feedback filter 16. The input to the forward filter 12 is the received data which includes the transmitted symbol sequence a_k, noise n_k and multipath h_i. The input to the feedback filter is the quantized equalizer output $\hat{a}_k$. The output of both the sections are summed 18 to form the final equalizer output $\tilde{a}_k$ 19 which is also the input to the next stage in a trellis-coded system, the trellis decoder. While a DFE performs better than a linear equalizer in severe ISI, the performance is limited by error propagation through the feedback filter 16 of the DFE 10. Error propagation occurs in the feedback filter 16 when the quantized equalizer output $\hat{a}_k$ is not the same as the transmitted symbol a_k. If an error is made in determining the symbol $\hat{a}_k$ at the output of the slicer 14, this incorrect symbol is fed back to the input of the feedback filter 16 and propagates. As known, the slicer 14 quantizes the filtered signal, providing an estimate of the symbol received. In many systems which employ error correction codes like trellis codes and/or Reed-Solomon codes to obtain very low error rates at moderate SNRs, the "raw" symbol error rate (SER) at the equalizer output can be extremely high. For example, in a Vestigal Sideband (VSB) system, at threshold in white noise the SER at the equalizer output is about 0.2. The increased error propagation due to these high SERs can cause the DFE to lose a couple of db in performance as compared to the case

-2-

of no error propagation. Additionally, the error
propagation causes the error sequence at the equalizer
output to be correlated, since it depends on past incorrect
symbol decisions. This correlation has an adverse effect
on the subsequent trellis decoder which is usually designed
for a white noise sequence.

According to the IEEE 802.11b high-rate wireless
communications standard implementing direct sequence spread
spectra techniques, bit stream data may be encoded using a
standard known as Complementary Code Keying (CCK). This
CCK encoding scheme is used to achieve 5.5 Mbps or 11 Mbps
in wireless LANs. Rather than using the Barker code, which
is the standard 11-bit chipping sequence used to encode
data bits, CCK requires that data be encoded using a series
of codes called Complementary Sequences. Because there are
256 unique code words that can be used to encode the
signal, up to 8 bits can be represented by any one
particular code word (assuming an 11 Mps bit stream).

It is the case that, for most symbol modulation
schemes, the Decision Feedback Equalizer adequately
performs notwithstanding the difficulty of providing symbol
estimates $\hat{a}_k$. This is because the estimation is done on a
symbol-by-symbol basis.

It would be highly desirable to provide a DFE
that exploits the fact that when a symbol is received,
there is a relationship between the other symbols before
and after it, e.g., if it is in the middle of a CCK code
word.

Past efforts have relied upon providing
equalization first and, then perform the CCK decoding.

-3-

However, it would be highly desirable to provide both CCK modulation decoding and equalization at the same time.

SUMMARY OF THE INVENTION

An object of this invention is to provide an improved system and method for decoding CCK-encoded digital data streams implementing a trellis-decoding technique.

Another object of the present invention is to provide an improved receiver device for use in digital communication systems implementing the IEEE 802.11b high rate digital communications standard that implements a novel, computationally efficient trellis decoding technique for decoding CCK encoded symbols.

These and other objectives are attained with a method and system for performing joint equalization and decoding of Complementary Code Key (CCK) encoded symbols. The system comprises: a decision feedback equalizer (DFE) structure for simulating an inverse communications channel response and providing an output comprising an estimation of the received symbols, the DFE structure including a forward equalizer path and a feedback equalizer path including a feedback filter; and, a CCK decoder embedded in the feedback path and operating in conjunction with a feedback filter therein for decoding the chips based on intermediate DFE outputs including those chips corresponding to past decoded CCK symbols. Decisions on a symbol chip at a particular time are not made until an entire CCK codeword that the chip belongs to is decoded, thereby reducing errors propagated when decoding the symbols.

-4-

Advantageously, the trellis decoding method is implemented as a computationally efficient 64-state trellis.

Further benefits and advantages of the invention will become apparent from a consideration of the following detailed description, given with reference to the accompanying drawings, which specify and show preferred embodiments of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 illustrates generally a DFE equalizer structure according to the prior art;

Figure 2 depicts the joint DFE and CCK decoding technique according to a first embodiment of the invention;

Figure 3 depicts the joint DFE and CCK decoding technique according to a preferred embodiment of the invention; and,

Figure 4 depicts the generated trellis structure underlying the joint DFE and CCK decoding technique of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention is directed to a digital communications system and a computationally efficient decoding structure for decoding data received in the form of symbols modulated according to Complementary Code Keying (CCK) technique. The system of the invention will be described herein for the case of an 11 Mbps digital data stream in accordance with the IEEE 802.11b standard, however, it is understood that skilled artisans may readily

-5-

apply the principles described herein to other bit stream data rates in accordance with the standard, e.g., 5.5 Mps.

In a first embodiment of the invention, as shown in Figure 2, the system includes a receiver device comprising, in part, a Decision Feedback Equalizer ("DFE") 20 such as the equalizer used in a 802.11b communications receiver. The DFE 20 may be a fractionally spaced decision-feedback equalizer (DFE) having a forward filter 12' with taps that are fractionally (T/2) spaced. This forward filter 12' will perform both matched filtering and equalization. The equalizer 20 further includes a feedback filter 16' that may be sample spaced, i.e., T spaced, where "T " denotes the sample rate, which is also the chip rate, e.g., 11MHz. The input to the equalizer 20 is assumed to be T/2 spaced, i.e., sampled at 22 MHz. The DFE 20 may be used for all the possible 802.11b communications modes, i.e., 1, 2, 5.5 and 11 Mbps. In the first embodiment, as shown in Figure 2, the input to the feedback filter section 16 comprises the output of the slicer 14', which provides an estimate of the true transmitted chips and may include either a BPSK or QPSK slicer, depending on the transmitted mode. Equation (1) describes this structure as follows:

$$\widetilde{c}_k = \sum_{i=0}^{L_f-1} f_i r_{2k+d_f-i} + \sum_{i=1}^{L_b} b_i \hat{c}_{k-i} \qquad (1)$$

where, f_i are the forward equalizer taps, b_i are the feedback equalizer taps, r_k is the received input stream at T/2 rate, $\widetilde{c}_k$ is the DFE equalizer output at T rate, L_f is

-6-

the length of the forward filter, d_f is the delay through
the forward filter, L_b is the length of the feedback
filter, and $\hat{c}_k$ is the slicer output which is an estimate of
the true transmitted chip c_k. As shown in Figure 2, there
is provided a CCK decoder 25 for providing the decoding of
the received chips. It should be understood that in other
embodiments, the decode element 25 may comprises a typical
Barker despreader, for the low-rate modes. The input to
the CCK decoder 25 is $\tilde{c}_k$. It is readily seen that, in this
embodiment, equalization and CCK decoding/Barker
dispreading are completely separated and, as such, may be
subject to propagation errors due to errors made by the
slicer.

For improved performance, in a preferred
embodiment, the DFE structure 20' illustrated in Figure 3
is implemented. In accordance with the configuration
illustrated in Figure 3, the CCK decoder/Barker despreader
device 25' is embedded into a DFE feedback loop 30
including the feedback filter. Decoding and equalization
is done on a block of eight chips for the CCK modes and on
eleven chips for DSSS modes. Equation 2) describes the
structure for the CCK mode as follows:

$$
\begin{aligned}
\tilde{c}_{k+j} &= \sum_{i=0}^{L_f-1} f_i r_{2(k+j)+d_f-i} + \sum_{i=1}^{L_b} b_i \tilde{c}_{k+j-i}, \; j=0,1,\ldots 7 \\
&= \sum_{i=0}^{L_f-1} f_i r_{2(k+j)+d_f-i} + \sum_{i=j+1}^{L_b} b_i \hat{c}_{k+j-i} + \sum_{i=1}^{j} b_i c_{k+j-i} \\
&= s_{k+j} + \sum_{i=1}^{j} b_i c_{k+j-i} \quad\quad\quad\quad\quad (2)
\end{aligned}
$$

-7-

where $s_{k+j} = \sum_{i=0}^{L_f-1} f_i r_{2(k+j)+d_f-i} + \sum_{i=j+1}^{L_b} b_i \hat{c}_{k+j-i}$ $j=0,\ldots 7$ represents the

intermediate DFE equalizer outputs that include, in the

feedback filter, only those chips corresponding to past

decoded CCK symbols and the $\sum_{i=1}^{j} b_i c_{k+j-i}$ component represents

the chips comprising the present transmitted symbol.

According to the invention, the CCK decoder 20' then

chooses, from the set of 256 possible codewords, that

codeword $[c_0, c_1, \ldots, c_7]$ that minimizes the metric set forth

in equation 3) as follows:

$$\sum_{j=0}^{7} \left| s_{k+j} + \sum_{i=1}^{j} b_i c_{j-i} - c_j \right|^2 \qquad (3)$$

It is understood that similar equations may be written for

the DSSS modes except that blocks of 11 chips at a time

would be considered and that there are only 2, or 4

possible 11-chip words.

The configuration in accordance with the

preferred embodiment greatly reduces error propagation, as

decisions on the chip at time k are not made until the

entire CCK codeword that the chip belongs to is decoded.

The complexity of this solution is greater than the

configuration in accordance with the first embodiment

(Figure 2), however, according to the preferred embodiment

of the invention, a computationally efficient decoding

method employing a trellis structure is provided, as will

be described.

-8-

According to this method, the variable $\underline{c} = [c_0,$ $c_1, \ldots, c_7]$ represents the 8-symbol CCK codeword. The symbols in the codeword "$\underline{c}$" are expressed in terms of the four QPSK phases $\varnothing_1$, $\varnothing_2$, $\varnothing_3$ and $\varnothing_4$ used to create the CCK codes according to equation 4) as follows:

$$\underline{c} = \left[e^{j(\phi1+\phi2+\phi3+\phi4)},e^{j(\phi1+\phi3+\phi4)},e^{j(\phi1+\phi2+\phi4)},-e^{j(\phi1+\phi4)},e^{j(\phi1+\phi2+\phi3)},e^{j(\phi1+\phi3)},-e^{j(\phi1+\phi2)},e^{j(\phi1)}\right] \quad (4)$$

As the phase $\varnothing_1$ is common to all the symbols in a codeword, the following definitions for values α_1, α_2 and α_3 are provided according to equation 5) as follows:

$$\alpha_1 = \varnothing_2 + \varnothing_3 + \varnothing_4$$
$$\alpha_2 = \varnothing_3 + \varnothing_4$$
$$\alpha_3 = \varnothing_2 + \varnothing_4 \quad (5)$$

Thus, the CCK codeword $\underline{c}$ may be re-written in terms of the variables α_i and $\varnothing_1$ according to equation 6) as follows:

$$\underline{c} = e^{j\phi1}\left[e^{j\alpha1},e^{j\alpha2},e^{j\alpha3},-e^{j(\alpha2+\alpha3-\alpha1)},e^{j(2\alpha1-\alpha2-\alpha3)},e^{j(\alpha1-\alpha3)},-e^{j(\alpha1-\alpha2)},1\right] \quad (6)$$

In other words, the codeword $\underline{c}$ may be expressed as: $\underline{c} = e^{j\phi1}\underline{d}$, where $\underline{d}$ is a function of α_1, α_2 and α_3 as shown in equation 6). Each of the α_i may take on one of 4 values $[0, \pi/2, \pi, 3\pi/2]$ and hence $\underline{d}$ belongs to a set of 64 possible vectors, whereas $\underline{c}$ can have 256 possible values. Utilizing a brute force methodology, the embedded CCK decoder may be programmed to choose from the set of 256

-9-

possible code words (a number 8 symbols long), that
minimizes the metric, and use the corresponding c_0, c_1,
..., c_7 values. For this brute force minimization, this
value would be computed 256 times (for each of the 256
possible combinations that could be transmitted) with the
combination that minimizes this metric distance being
picked.

According to the preferred embodiment, however,
rather than a brute force methodology, a trellis structure
may be used because of the memory effect of the feedback
filter in the DFE feedback loop 30 (Figure 3). That is,
the eight intermediate outputs s_{k+j}, j = 0, 1, ...,7, are
processed by the trellis to determine the transmitted
codeword at time k. Advantageously, as will be described,
the dimensionality of a trellis search may be reduced from
256 to 64. That is, as shown in Figure 4, a trellis
structure 100 is generated which represents basically as a
state diagram having an initial state $102_{j=0}$, whereby, in
the case of a multi-path channel and considering only the
contribution in the feedback filter from symbols in the
present CCK code word, the maximum number of states 102 in
a corresponding set $102_{j=0}$, ..., $102_{j=7}$ at each corresponding
level $103_{j=0}$, ..., $103_{j=7}$ (equal to eight (8) levels), grows up
to a maximum of sixty-four (64). Preferably, the trellis
structure is embodied as an algorithm executing in hardware
provided in the combined CCK decoder/equalizer feedback
filter structure (Figure 3), however, it could easily be
executed in software.

US020010 (702518)

In the description of how the programmed trellis structure and algorithm of Figure 4 operates to process the block of eight symbols s_{k+j}, reference is had to equation 3, which sets forth the metric to be minimized and which may be re-written in terms of variables $\underline{d}$ and $\varnothing_1$ according to equations 7) and 8) as follows:

$$\sum_{j=0}^{7} \left| s_{k+j} + \sum_{i=0}^{j} b_i c_{j-i} \right|^2 \tag{7}$$

where $b_0 = -1$; and, using the relation $c_i = e^{j\phi 1} d_i$, the following equation 8) is obtained:

$$\sum_{j=0}^{7} \left| s_{k+j} + e^{j\phi 1} \sum_{l=0}^{j} b_i d_{j-i} \right|^2$$

$$= \sum_{j=0}^{7} \left[\left| s_{k+j} \right|^2 + \left| \sum_{i=0}^{j} b_i d_{j-i} \right|^2 + 2 \operatorname{Re} \left(s^*_{k+j} e^{j\phi 1} \sum_{i=0}^{j} b_i d_{j-i} \right) \right] \tag{8}$$

where Re denotes a real part and * denotes complex conjugate. Minimizing the above equation is equivalent to minimizing the metric according to equation 9) as follows:

$$\sum_{j=0}^{7} \left| \sum_{i=0}^{j} b_i d_{j-i} \right|^2 + 2 \operatorname{Re} \left[e^{j\phi i} \sum_{j=0}^{7} s^*_{k+j} \sum_{i=0}^{j} b_i d_{j-i} \right] \tag{9}$$

Defining now the term $\chi_j = \sum_{i=0}^{j} b_i d_{j-i}$, then, the metric to be minimized may be set forth according to equation 10) as follows:

-11-

$$\sum_{j=0}^{7} |\chi_j|^2 + 2\operatorname{Re}\left[e^{j\phi 1} \sum_{j=0}^{7} s^*_{k+j} \chi_j\right]$$ (10)

Returning to Figure 4, a state in the trellis 100 is defined by the vector $[\alpha_1, \alpha_2, \alpha_3]$ and a block of eight (8) symbols s_{k+j}, $j = 0, 1, \ldots, 7$, are processed by the trellis as follows: At each time, j, the following quantities are calculated for each state 102 in the set $102_{j=0}, \ldots, 102_{j=7}$ as follows: χ_j, a real-valued $m_1(j) = |\chi_j|^2$ and a complex-valued $m_2(j) = s^*_{k+j}\chi_j$. Substituting these values into equation 10), the metric to be minimized may now be set forth according to equation 11) as follows:

$$\sum_{j=0}^{7} m_1(j) + 2\operatorname{Re}\left[e^{j\phi 1} \sum_{j=0}^{7} m_2(j)\right]$$ (11)

Thus, for every branch in a trellis path, two (2) quantities need to be computed: a real-valued $m_1(j) = |\chi_j|^2$ and a complex-valued $m_2(j) = s^*_{k+j}\chi_j$. These quantities are then added to the corresponding quantities of the state from which the branch originated. Unlike the trellis decoding in a convolutional code, here there is only one branch coming into any state and hence there is no "survivor path". At $102_{j=0}$ there are 4 possible paths corresponding to the 4 possible values of α_1, at $102_{j=1}$ there are 16 possible paths corresponding to the 16 possible combinations of $[\alpha_1, \alpha_2]$, at $102_{j=2}$ there are 64 possible paths corresponding to the 64 possible combinations of $[\alpha_1,$

-12-

α_2, α_3] and after that the trellis size does not grow since all succeeding values of d_j are functions of the same three phases, as shown in equation (6). After the entire codeword has been received, i.e., at the end of the trellis 100 when $102_{j=7}$ at level $103_{j=7}$, for each of the 64 states the metric set forth in equation 11) is calculated for each of the 4 possible values of $\emptyset_1$, for a total of 256 values. Then, the state and $\emptyset_1$ value is chosen that corresponds to the minimum metric. The vector [α_1, α_2, α_3] corresponding to the state with the minimum metric along with the $\emptyset_1$ value is then used to calculate the transmitted codeword c.

It should be understood that the extra quantity $m_1(j)$ has to be computed in the equation 11), because, in general, this term will be a function of the codeword and the filter taps. In the absence of multipath, i.e., $b_0 = -1$ and $b_j = 0$ elsewhere, it is easy to see that $m_1(j) = 1$ always, and hence does not contribute towards the final metric.

As shown in Figure 4, at each state, the trellis structure moves to four (4) other values, depending upon what the s_0 value was because that goes into b_0 (equation 3). Then from each value, the trellis may branch into four other values. It is readily seen that, if at every state 102a,. . .,102n in the trellis there may be four possible inputs corresponding to four different values, then the structure would grow exponentially and basically end up with a number of combinations (states) equal to 4^8. However, according to the CCK structure, this trellis structure only moves to 64 states (i.e., 102c,...,102n) and then saturates. This is because all of the eight symbols

-13-

may be expressed in terms of just three (phase) values α_1, α_2 and α_3, i.e., each of these three phases can take one of four values so there is $4^3 = 64$ possible combinations which makes the trellis structure extremely manageable.

While it is apparent that the invention herein disclosed is well calculated to fulfill the objects stated above, it will be appreciated that numerous modifications and embodiment may be devised by those skilled in the art, and it is intended that the appended claims cover all such modifications and embodiments as fall within the true spirit and scope of the present invention.

-14-